

Contao Backup

14.9.2014, Andreas Fieger

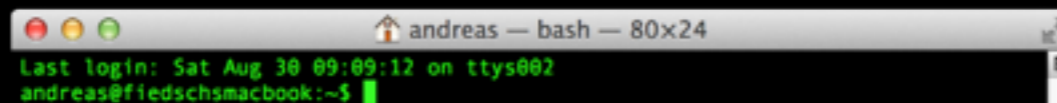
„Das ist doch ganz einfach“

Was brauchen wir also?

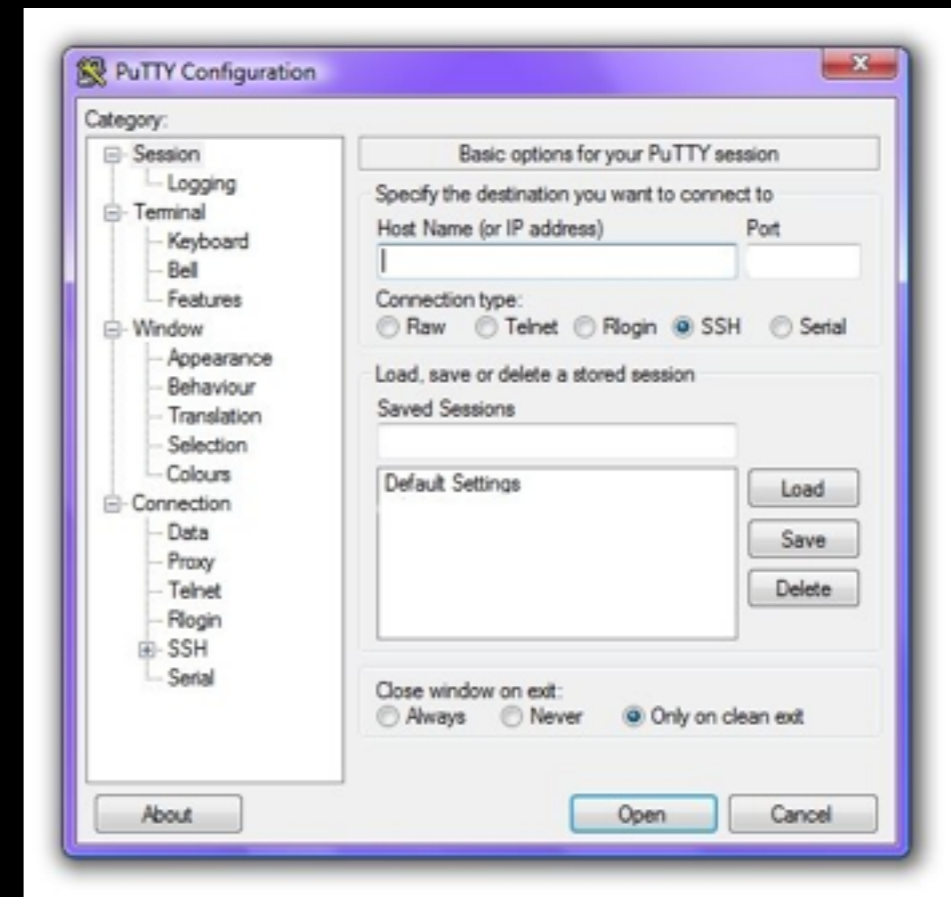
- Shell-Zugang (ssh, Linux, Mac: Terminal, Windows: Putty)
- Auf dem Server installierte Programme (mysqldump, mysql)
- Für eine Automatisierung cron jobs

Shell-Zugang

- `ssh user@host`
- `ssh -l user host`
- Putty Konfiguration ausfüllen und klicken

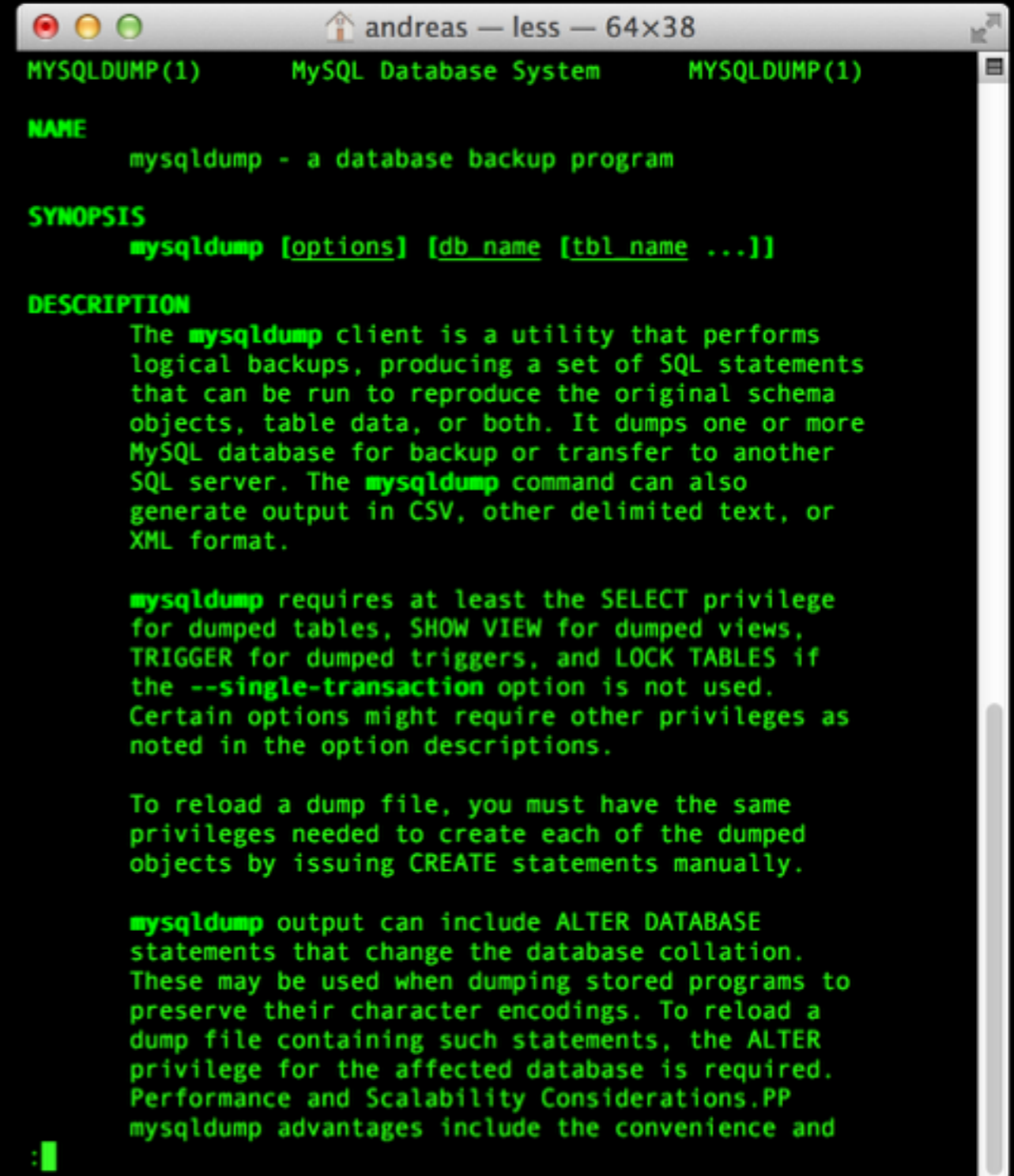


A terminal window titled "andreas — bash — 80x24" showing a successful SSH login. The output is: "Last login: Sat Aug 30 09:09:12 on ttys002" followed by the prompt "andreas@fiedschsmacbook:~\$".



Datenbank sichern: mysqldump

- `man mysqldump`



```
andreas — less — 64x38
MYSQLDUMP(1)      MySQL Database System      MYSQLDUMP(1)

NAME
  mysqldump - a database backup program

SYNOPSIS
  mysqldump [options] [db_name [tbl_name ...]]

DESCRIPTION
  The mysqldump client is a utility that performs logical backups, producing a set of SQL statements that can be run to reproduce the original schema objects, table data, or both. It dumps one or more MySQL database for backup or transfer to another SQL server. The mysqldump command can also generate output in CSV, other delimited text, or XML format.

  mysqldump requires at least the SELECT privilege for dumped tables, SHOW VIEW for dumped views, TRIGGER for dumped triggers, and LOCK TABLES if the --single-transaction option is not used. Certain options might require other privileges as noted in the option descriptions.

  To reload a dump file, you must have the same privileges needed to create each of the dumped objects by issuing CREATE statements manually.

  mysqldump output can include ALTER DATABASE statements that change the database collation. These may be used when dumping stored programs to preserve their character encodings. To reload a dump file containing such statements, the ALTER privilege for the affected database is required. Performance and Scalability Considerations.PP
  mysqldump advantages include the convenience and
```

mysqldump und mysql

Grundprinzip

```
mysqldump datenbank > datei.sql
```

```
mysql datenbank < datei.sql
```

Aufruf (erweitert)

weil „Das ist doch ganz einfach“ war natürlich ein Bischen gelogen ;-)

```
mysqldump -u user -p password -h host \  
--add-drop-table \  
--default-character-set utf8 \  
--skip-extended-insert \  
--hex-blob \  
datenbank \  
> dump.sql
```

„Aber ich habe gar keinen Shell-Zugang“

- Doch! (zumindest lokal ;-)
- Was kostet das nächst größere Paket vs. was kostet der Aufwand
- Backup und Restore == Umzug
(Entwicklungsrechner \Leftrightarrow Server)

Wir schauen in das Skript

```
#!/bin/bash
#
# Time-stamp: <2014-08-31 09:42:26 andreas>
#
# Backup von Contao Installationen (Dateien und Datenbank)
#
# - - - - -
# Konfiguration
# - - - - -
#
# Zielverzeichnis (wo sollen die erzeugten Backups liegen)
#
TARGET_DIR=/Users/andreas/Desktop
#
# Stammverzeichnis der Contao Installation
#
WEB_ROOT=/Users/andreas/Sites
#
# Name der Contao Installation
#
CONTAO_DIR=contao3.2
#CONTAO_DIR=contao_ifp
#
# - - - - -
# Ende Konfiguration
# - - - - -
```


Anmerkungen/Ausblick

- <https://gist.github.com/fiedsch/>
- Lokal: Installtool oder localconfig.php anpassen
- Cron Job
- Server mounten
 - `sshfs user@host:/remote/path /local/path`
 - `fusermount -u /local/path`
- Befehl(e) in PHP Datei
 - `(exec() oder shell_exec())`
 - Symfony Process Component